Deep Extended Hazard Models for Survival Analysis

Anonymous Author(s)
Affiliation
Address
email

Abstract

Unlike standard prediction tasks, survival analysis requires modeling right censored data, which must be treated with care. While deep neural networks excel in traditional supervised learning, it remains unclear how to best utilize these models in survival analysis. A key question asks which data-generating assumptions of traditional survival models should be retained and which should be made more flexible via the function-approximating capabilities of neural networks. Rather than estimating the survival function targeted by most existing methods, we introduce a Deep Extended Hazard (DeepEH) model to provide a flexible and general framework for deep survival analysis. The extended hazard model includes the conventional Cox proportional hazards and accelerated failure time models as special cases, so DeepEH subsumes the popular Deep Cox proportional hazard (DeepSurv) and Deep Accelerated Failure Time (DeepAFT) models. We additionally provide theoretical support for the proposed DeepEH model by establishing consistency and convergence rate of the survival function estimator, which underscore the attractive feature that deep learning is able to detect low-dimensional structure of data in high-dimensional space. Numerical experiments also provide evidence that the proposed methods outperform existing statistical and deep learning approaches to survival analysis.

1 Introduction

Across areas such as biomedical science and reliability engineering, survival data analysis is critically used to study the time until certain events occur (e.g. patient death in clinical applications, component failure in industrial applications). A key issue stems from the fact that many occurrence-times fail to be recorded due to natural data censoring. For example, a patient may relocate and drop out of a longitudinal clinical study, leading to the loss of follow-up observations for the remaining period of the study. For such right censored data, we only know that the true event time is greater than the last available observation time for this individual (but not by how much), and thus standard regression approaches are not applicable to most survival data. A naive approach that removes all censored individuals from the data would induce bias, as these tend to be individuals with greater event times.

Learning the distribution of the event time in the presence of censorship is a fundamental task in survival analysis. A wide array of statistical methodologies have been developed to study this distribution. Among them, Kaplan and Meier [31] considered a product of successive conditional probabilities to estimate the *survival function*, which encodes tail probabilities of the event time distribution. As a nonparametric approach, the Kaplan-Meier estimator assumes no particular form for either the underlying distribution of event time or the relationship between individuals' observed feature-values and the outcome event time. Alternatively, the Nelson-Aalen estimator considers the cumulative hazard function from a counting process perspective [1, 42]. The classical Nelson-Aalen estimator was improved by leveraging more powerful supervised learning methods, in particular random forests, which lead to the *random survival forest* model [27].

Submitted to 35th Conference on Neural Information Processing Systems (NeurIPS 2021). Do not distribute.

While the Kaplan-Meier estimator has been extended to assess the effect of multiple subject features (covariates) on the associated event time [4], it struggles to work with even moderately high-dimensional features. A semi-parametric approach, the Cox proportional hazards (CoxPH) model [12], has thus been popularly adopted as a dimension reduction approach. However, the CoxPH model is restrictive as it assumes that the conditional log hazard functions for individuals are parallel to each other. When this assumption fails to be realistic, the accelerated failure time (AFT) model [29, 55] often emerges as an alternative, which simply assumes a linear regression model for the log event time against the individuals' feature-values. Like the CoxPH model, the AFT model is also semi-parametric because the error distribution in the regression model is unspecified. An even simpler approach is to adopt a parametric approaches for survival data by assuming that the distribution of event time obeys a certain known law, such as the Weibull or Gamma distribution [38]. While non-parametric methods are more flexible, a semi-parametric or parametric approach can be more effective when there are limited data, if the distribution of event times can be closely approximated by a certain semi-parametric or parametric form.

Rapid advances in data collection and analysis have encouraged researchers to explore the potential of deep learning in the field of survival analysis. Faraggi and Simon [16] proposed to replace the linear predictor of the CoxPH model with a single hidden-layer neural network, but this extension failed to provide reliable improvements [56]. Under the same CoxPH framework, Katzman et al. [32] and Kvamme et al. [35] applied neural networks of greater complexity to obtain more flexible nonlinear models. Zhu et al. [58] and Tarkhan et al. [50] used convolutional architectures to explore the relationship between event time and unstructured features like images, and domain-specific variants of these models have been developed for particular applications, such as genomic data [8, 21], clinical research [43], pedestrian waiting time in urban areas [28] and heart failure rehospitalization [30].

Beyond the CoxPH framework, deep learning has been used to model survival data in various alternative forms, including: neural network versions of the Beta-Logistic model [26], recurrent neural networks applied with time-discretization [18, 45], generative adversarial modeling of time-to-event distributions [6], generalized forms of regression [34, 37], and more flexible extensions of the Kaplan-Meier estimator [7, 49] and the Weibull distribution [44]. Among existing deep learning methods for survival analysis, the distribution free approaches [7, 10, 23, 34, 37] make almost no assumptions on the underlying survival function structure, but these very general models may exhibit poor sample complexity, requiring very large data to accurately estimate the survival function.

In this paper, we adapt neural networks to generalize extended hazard (EH) model [9], which includes both the CoxPH and AFT models as special cases. The proportional hazards assumption are often violated in practice and can lead to misinterpretation of the results, especially when there is heterogeneity among individuals [2]. In contrast, the AFT framework directly considers the relationship between feature values and the event time, which results in a time-dependent hazard ratio that is more realistic. Keiding et al. [33] also show that, relative to CoxPH, AFT models are more stable when accounting for omitted features. Other studies of CoxPH and AFT models have however reported conflicting results [40], and thus practitioners may find it difficult to choose which model to use in their applications. The EH model [15, 51, 52], under which feature values can affect both the baseline hazard rate and hazard ratio, flexibly combines the CoxPH and AFT models into a more general framework. EH models also subsume a wider range of survival models such as Weibull and Gamma distributions. Compared with the current practice of deep learning under the CoxPH or AFT framework, our deep extensions of EH model retain the advantages of standard EH model over CoxPH and AFT, and achieve better empirical performance on real survival data.

While deep learning has enjoyed a number of breakthroughs in survival data analysis, mathematical understanding of its success is still lagging far behind. In this paper, we provide theoretical support for Deep Extended Hazard (DeepEH) model, which replaces the two linear risk predictors by two nonparametric functions (cf. h_1 and h_2 in (4)). We leverage the advantages of neural networks and show that the resulting estimators of h_1, h_2 and survival functions are asymptotically consistent and enjoys fast convergence rates. Specially, the convergence rates are determined by the intrinsic dimension of the underlying functions rather than the original high-dimension input features. These profound results reveal that neural networks are able to identify low-dimensional structure of the data, hence avoid the curse of dimensionality. This, to some extent, explains the great success of deep learning and distinguishes our contributions from existing work.

2 Background

In survival analysis, we are interested in modeling the time T until some event of interest (i.e. survival time), whose probability distribution is typically characterized in one of four equivalent ways: probability density function, survival function, hazard function, or cumulative hazard function. The *survival* function $S(t)$ is the probability of a individual surviving beyond time t:

$$S(t) = \mathbb{P}(T > t) = 1 - \int_0^t f(s)ds,$$

where f is the probability density function of survival times. The *hazard* function, denoted by $\lambda(t)$, characterizes the chance that an event occurs in an infinitesimal interval after time t, given it has not yet occurred at time t:

$$\lambda(t) = \lim_{\delta \to 0} \frac{\mathbb{P}(t \leq T < t + \delta | T \geq t)}{\delta}.$$

This results in the relationship: $S(t) = \exp\{-\Lambda(t)\}$, where $\Lambda(t) = \int_0^t \lambda(s)ds$ is the *cumulative hazard* function.

In practice, a survival study of n subjects will contain less than n completely observed event times due to right censoring. For instance, the event of some individuals may not have occurred yet at the time when the study is concluded or the subject may have dropped out of the study. For the i-th subject in the study, T_i and C_i denote, respectively, the event time and the potential censoring time; and $X_i \in \mathbb{R}^p$ denotes the observed features (covariates). Thus, the observed data from a typical survival study contain independent observations $\mathcal{D} = \{X_i, O_i, \Delta_i\}_{i=1}^n$, where the observed time $O_i = \min(T_i, C_i)$ and the event indicator $\Delta_i = 1$, if the observed time is T_i, i.e. $T_i \leq C_i$, otherwise $\Delta_i = 0$, the subject is censored at C_i. Given the feature values X_i, the event time T_i and censoring time C_i are typically assumed to be independent. We aim to estimate the survival function and hazard function using censored data in which the true event times may be unavailable. Conditional on the features X, the overall log-likelihood of the censored survival data $\mathcal{D}$ is:

$$\mathcal{L}_n = \frac{1}{n} \sum_{i=1}^n \Big[\Delta_i \log f(O_i|X_i) + (1 - \Delta_i) \log S(O_i|X_i)\Big] = \frac{1}{n} \sum_{i=1}^n \Big[\Delta_i \log \lambda(O_i|X_i) - \Lambda(O_i|X_i)\Big]. \tag{1}$$

Here, $f(t|X)$, $S(t|X)$, $\lambda(t|X)$ and $\Lambda(t|X)$ denote the conditional density, survival, hazard, and cumulative hazard function of $T|X$, respectively. To obtain maximum likelihood estimates, one can assume that the conditional distribution of T follows a particular parametric distribution. However to avoid restrictive assumptions, it is advantageousß to use more flexible alternatives such as the CoxPH and AFT models.

2.1 Cox Proportional Hazard Model

Given features X, the Cox proportional hazard (CoxPH) model [11] for event time T, is a semi-parametric model whose hazard function has a multiplicative form in terms of a baseline hazard function $\lambda_0(t)$ and an exponential risk function of X:

$$\lambda(t|X) = \lambda_0(t) \exp(\beta^\top X),$$

where the baseline hazard function $\lambda_0(t)$ and parameter β are both unknown. The function $\lambda_0(t)$ represents the baseline hazard rate when all features take the baseline value of zero. It is difficult to estimate λ_0 and β directly from the likelihood function (1). Luckily the partial likelihood method [11, 12] has been developed to handle the parametric component of the model without involving the nonparametric component $\lambda_0(t)$. Specifically, the partial likelihood technique first estimates β by maximizing the log-partial likelihood:

$$\mathcal{L}_n(\beta) = \frac{1}{n} \sum_{i=1}^n \Delta_i [\beta^\top X_i - \log(\sum_{j=1}^n Y_j(O_i) e^{\beta^\top X_j})],$$

where $Y_j(t) = 1(O_j \geq t)$. Intriguingly, the log-partial likelihood does not involve the nonparametric baseline hazard function λ_0, which is a major advantage in accomplishing dimension reduction. Once the parameter β has been estimated through the log partial likelihood, the cumulative baseline hazard function $\Lambda_0(t) = \int_0^t \lambda_0(s)ds$ can be estimated through the Breslow estimator [5]. This is the main reason why the Cox proportional hazards model dominates the field of survival analysis for nearly fifty years.

2.2 Accelerated Failure Time

The accelerated failure time (AFT) model [55] is an alternative to CoxPH when the proportional hazards (PH) assumption is violated. Given feature values X, the hazard function for an AFT model is of the form:

$$\lambda(t|X) = \lambda_0(te^{\beta^\top X})e^{\beta^\top X}, \tag{2}$$

where λ_0 is an unknown baseline hazard function and β is an unknown parameter. Unlike the CoxPH, the AFT model allows the feature X to influence the time curve of the baseline hazard function as well. AFT with $\beta > 0$ aasumes that a subject with a larger X value will have double jeopardy through a higher risk factor $e^{\beta^\top X}$ and a faster hazard schedule $\lambda_0(te^{\beta^\top X})$ compared to the baseline hazard $\lambda_0(t)$, hence the name "accelerated failure".

2.3 Extended Hazard Model

In practice, it may be unclear whether a CoxPH or AFT model is more suitable for the data in hand. The Extended Hazard (EH) model [9] includes both the CoxPH and AFT model as special cases, thereby providing a more flexible and larger class of models to use in such practical applications. While AFT (2) allows for only one feature effect β, the EH model extends this is to allow two different feature effects β_1 and β_2:

$$\lambda(t|X) = \lambda_0(te^{\beta_1^\top X})e^{\beta_2^\top X}. \tag{3}$$

Note that the EH model collapses to the CoxPH model when $\beta_1 = 0$, and the AFT model when $\beta_1 = \beta_2$. The EH model can thus flexibly leverage the complementary strengths of the CoxPH and AFT model, adaptively exploiting whichever assumptions are more appropriate for a given dataset.

3 Methods

3.1 Deep Extended Hazard Model

Our DeepEH model is derived by extending the standard EH model in (3) to the following hazard function:

$$\lambda(t|X) = \lambda_0(te^{h_1(X)})e^{h_2(X)} \tag{4}$$

where $h_1(X)$ and $h_2(X)$ are unknown functions and λ_0 is an unknown and non-negative baseline hazard function. When $h_1 = 0$, it collapses to the popular DeepSurv [32]. When $h_1 = h_2$, we call this the Deep Accelerated Failure Time (DeepAFT) model as it is the deep version of the AFT model (2). Interestingly, DeepAFT has not been explored much in the literature. Our theory and method for DeepEH extends to DeepAFT as well. Let $\Lambda_0(t) = \int_0^t \lambda_0(s)ds$ be the cumulative baseline hazard function. The survival function of DeepEH can be expressed as:

$$S(t|X) = \exp\{-\Lambda_0(te^{h_1(X)})e^{h_2(X)-h_1(X)}\}. \tag{5}$$

Analogous to [52], we propose a pseudo-likelihood function for the DeepEH model:

$$\begin{aligned}\mathcal{L}_n(h_1, h_2) =& \frac{1}{n}\sum_{i=1}^n \Delta_i[h_2(X_i) - R_i(h_1)] + \frac{1}{n}\sum_{i=1}^n \Delta_i \log\Big[\frac{1}{nb}\sum_{j=1}^n \Delta_j K\Big(\frac{R_j(h_1) - R_i(h_1)}{b}\Big)\Big] \\ &- \frac{1}{n}\sum_{i=1}^n \Delta_i \log\Big[\frac{1}{n}\sum_{j=1}^n \frac{e^{h_2(X_j)}}{e^{h_1(X_j)}}\Phi\Big(\frac{R_j(h_1) - R_i(h_1)}{b}\Big)\Big],\end{aligned}$$

where $K(\cdot)$ is a known kernel function, $R_i(h_1) = \log(O_i e^{h_1(X_i)})$, b is a positive bandwidth and $\Phi(t) = \int_{-\infty}^t K(s)ds$.

We use two neural networks to approximate the unknown functions $h_1(X)$ and $h_2(X)$ in (4). More specifically, we learn:

$$(\hat{h}_1, \hat{h}_2) = \underset{g_1, g_2 \in \mathcal{G}}{\arg\min}\{-\mathcal{L}_n(g_1, g_2) + \rho J(g_1, g_2)\}$$

where $\mathcal{G}$ again denotes a certain family of NNs and J is a regularization term weighted by $\rho \geq 0$ and it is set to be L^2 regularization for the weights and biases of g_1 and g_2 in this paper. We estimate respectively the baseline hazard and cumulative baseline hazard functions via:

$$\hat{\lambda}_0(t) = \frac{\frac{1}{nt}\sum_{i=1}^n \Delta_i K_b(R_i(\hat{h}_1) - \log t)}{\frac{1}{n}\sum_{i=1}^n \frac{e^{\hat{h}_2(X_i)}}{e^{\hat{h}_1(X_i)}} \int_{-\infty}^{R_i(\hat{h}_1)-logt} K_b(s)ds} \text{ and } \hat{\Lambda}_0(t) = \int_0^t \hat{\lambda}_0(s)ds.$$

Based on (5), we further estimate the survival function as:

$$\hat{S}(t|X) = \exp\{-\hat{\Lambda}_0(te^{\hat{h}_1(X)})e^{\hat{h}_2(X)-\hat{h}_1(X)}\}. \tag{6}$$

3.2 Theoretical Analysis

To study the asymptotic properties of our estimators, we assume our multilayer feedforward neural networks follow a special structure with parameters $\boldsymbol{d}, s, K$ and A described below. Recall that a K-layer neural network with parameter $\boldsymbol{d} = (d_0, \cdots, d_{K+1})$ is a composition of functions:

$$g(x) = W_K g_{K-1}(x) + \mu_K, g_{K-1}(x) = \sigma(W_{K-1}g_{K-2}(x) + \mu_{K-1}), \cdots, g_0(x) = \sigma(W_0 x + \mu_0)$$

with activation function σ, weight matrices $W_k \in \mathbb{R}^{d_{k+1}\times d_k}$ and bias vectors $\mu_k \in \mathbb{R}^{d_k}$ for $k = 0, \cdots, K$. Denote $\|\cdot\|_\infty$ the supremum norms of matrices or vectors, we define

$$\mathcal{G}(K,\boldsymbol{d}) = \{g : g \text{ is } K\text{-layer neural networks with parameter } \boldsymbol{d} \text{ and } \max_{k=1,\cdots,K} \|W_k\|_\infty \vee \|\mu_k\|_\infty \leq 1\},$$

where $a \vee b = \max(a, b)$. Neural networks used in practice often contain many parameters, which can lead to overfitting. To address this, [48] proposed dropout regularization, which randomly sets some of network's activations to zero during training. Inspired by dropout, we further define

$$\mathcal{G}(\boldsymbol{d}, s, K, A) = \{g \in \mathcal{G}(K, \boldsymbol{d}) : \sum_{k=1}^K \|W_k\|_0 + \|\mu_k\|_0 \leq s, \text{and } \|g\|_\infty \leq A\}, \tag{7}$$

where $\|\cdot\|_0$ is the number of non-zero entries of matrix/vector and $\|g\|_\infty$ is the uniform norm of function g.

To obtain asymptotic results for neural networks, we consider a Hölder class of smooth functions with parameters $\alpha, B > 0$ and domain $\mathbb{D} \subset \mathbb{R}^p$ defined as:

$$\mathcal{H}_p^\alpha(\mathbb{D}, B) = \{h : \mathbb{D} \to \mathbb{R} : \sum_{\beta:|\beta|<\alpha} \|\partial^\beta h\|_\infty + \sum_{\beta:|\beta|=\lfloor\alpha\rfloor} \sup_{x,y\in\mathbb{D},x\neq y} \frac{|\partial^\beta h(x) - \partial^\beta h(y)|}{\|x-y\|_\infty^{\alpha-\lfloor\alpha\rfloor}} \leq B\},$$

where $\lfloor\alpha\rfloor$ is the largest integer strictly smaller than α, $\partial^\beta := \partial^{\beta_1}\ldots\partial^{\beta_p}$ if $\beta = (\beta_1, \cdots, \beta_p)$, and $|\beta| = \sum_{k=1}^p \beta_k$. Examples of Hölder function include a wide classes of functions, such as polynomials, trigonometric functions, exponential and logarithmic functions. Especially, it also includes Lipschitz continuous functions when $\alpha = 1$.

For some $L \geq 0$ and $\boldsymbol{p} = (p_0, \ldots, p_{L+1}) \in \mathbb{N}^{L+2}$, we also consider composite Hölder function:

$$h = h_L \circ h_{L-1} \circ \cdots \circ h_1 \circ h_0$$

where $h_l : [a_l, b_l]^{p_l} \to [a_{l+1}, b_{l+1}]^{p_{l+1}}, l = 0, \cdots, L$. Denote $h_l = (h_{l1}, \cdots, h_{lp_{l+1}})^T$ and let p_{lj} be the unique number of features that each h_{lj} depends on. Define $p_l^* = p_{l1} \vee \cdots \vee p_{lp_{l+1}}$. For illustration, if

$$h(x_1, \ldots, x_9) = h_{11}(h_{01}(x_1, x_2, x_3), h_{02}(x_4, x_5, x_6), h_{03}(x_7, x_8, x_9)), \tag{8}$$

we have $p_0^* = p_1^* = 3$. Further, we assume $h_{lj} \in \mathcal{H}_{p_l^*}^{\alpha_l}([a_l, b_l]^{p_l^*}, B)$ and define the underlying function space

$$\begin{aligned}\mathcal{H}(L, \boldsymbol{p}, \boldsymbol{p}^*, \alpha, B) = \{h =& h_L \circ h_{L-1} \circ \cdots \circ h_1 \circ h_0 : h_l = (h_{l1}, \cdots, h_{lp_{l+1}})^T, \\ & h_{lj} \in \mathcal{H}_{p_l^*}^{\alpha_l}([a_l, b_l]^{p_l^*}, B), \text{ for some } |a_l| \vee |b_l| \leq B\},\end{aligned} \tag{9}$$

where $\boldsymbol{p} = (p_0, \cdots, p_{L+1})$, $\boldsymbol{p}^* = (p_0^*, \cdots, p_L^*)$ and $\alpha = (\alpha_0, \cdots, \alpha_L)$. For this function space, we denote $\alpha_l^* = \alpha_l \prod_{i=l+1}^{L} (\alpha_i \wedge 1)$, $l^* = \arg\min_{l=1,\ldots,L} \{\alpha_l^*/(2\alpha_l^* + p_l^*)\}$, and

$$r_n = n^{-\frac{\alpha_{l^*}^*}{2\alpha_{l^*}^* + p_{l^*}^*}}. \tag{10}$$

Here $a \wedge b = \min(a, b)$. Without loss of generality, we assume that each feature has been rescaled to the unit interval, $X \in [0, 1]^p$. To establish theoretical properties of the DeepEH model in (3), here we first introduce some main assumptions:

(A1) True functions h_1 and h_2 are elements of a Hölder class $\mathcal{H}_0 = \{h \in \mathcal{H}(L, \boldsymbol{p}, \boldsymbol{p}^*, \alpha, B) : \mathbb{E}[\Delta h(X)] = 0\}$.

(A2) The observation times O_i lie in a finite interval $[0, \tau]$, and the baseline hazard λ_0 in (4) is positive with twice continuous derivative in $[0, \tau)$.

(A3) The sub-density $p(t) := d\mathbb{P}(O \leq t, \Delta = 1)/dt$ is twice differentiable in $[0, \tau)$ and there exists positive constant c such that $\mathbb{P}(T \geq \tau | X) \geq c$ with probability one.

(A4) The kernel $K(\cdot)$ is twice-continuously differeantiable.

(A5) $K = O(\log n)$, $s = O(nr_n^2 \log n)$, $d_1 \wedge \ldots \wedge d_K = O(nr_n^2)$ and $d_1 \vee \ldots \vee d_K = O(n)$.

Assumption (A1) is required for the identifiability of h_1 and h_2. Assumptions (A2) and (A3) are common smoothness assumptions for right censored data [51], where the study ends at time τ. The second part of Assumption (A3) are usually satisfied in reality when the follow up study must end while some subjects are still alive at the end of the study. Assumption (A4) ensures that the likelihood function can be well approximated by the pseudo-likelihood function. Assumption (A5) configures the structure of neural networks in (7) that the neural network grows as sample size increases. Our main theoretical results are stated below and their proofs relegated to Appendix.

Theorem 1 *Under assumptions (A1)-(A5), if $\rho s \to 0$ as $n \to \infty$ and $b = O(n^{-1/5})$, for any $\epsilon > 0$, there exist an estimator $\hat{\boldsymbol{h}} = (\hat{h}_1, \hat{h}_2) \in \mathcal{G}(\boldsymbol{d}, s, K, A) \times \mathcal{G}(\boldsymbol{d}, s, K, A)$ for some sufficiently large $A > 0$, such that*

$$\lim_{n\to\infty} \mathbb{P}(d(\hat{\boldsymbol{h}}, \boldsymbol{h}) \geq \epsilon) = 0,$$

where $\boldsymbol{h} = (h_1, h_2)$ and $d(\hat{\boldsymbol{h}}, \boldsymbol{h}) = \|\hat{h}_1 - h_1\|_{L^2([0,1]^p)} + \|\hat{h}_2 - h_2\|_{L^2([0,1]^p)}$. Moreover, we have

$$\lim_{n\to\infty} \mathbb{P}(\|\hat{S}(\cdot|X) - S(\cdot|X)\|_{L^2([0,\tau])} \geq \epsilon) = 0.$$

From the proof of Theorem 1, we observe that both the approximation error and estimation error contributes to the error $\hat{\boldsymbol{h}} - \boldsymbol{h}$. The approximation error is the smallest distance between $\boldsymbol{h}$ and the neural networks space $\mathcal{G}(\boldsymbol{d}, s, K, A) \times \mathcal{G}(\boldsymbol{d}, s, K, A)$, for which more complex $\mathcal{G}(\boldsymbol{d}, s, K, A)$ yields smaller approximation errors. On the other hand, the estimation error is the difference between the minimizer $\hat{\boldsymbol{h}}$ and the optimal approximation of $\boldsymbol{h}$ in $\mathcal{G}(\boldsymbol{d}, s, K, A)$. With increasing sample sizes n, both the approximation error and estimation error become smaller. The next theorem shows the exact convergence rates of the proposed estimators.

Theorem 2 *Let $\alpha_* = \lfloor \alpha_{l^*}^* \rfloor + 1$. Under assumptions (A1)-(A5), if the sub-density $p(t)$ is α_*-th continuously differentiable, $b = O(n^{-1/(2\alpha_*+1)})$ and $\rho s = O(r_n^2)$, we have*

$$d(\hat{\boldsymbol{h}}, \boldsymbol{h}) = O_p(r_n \log^2 n) \quad \textit{and} \quad \|\hat{S}(\cdot|X) - S(\cdot|X)\|_{L^2([0,\tau])} = O_p(r_n \log^2 n).$$

Theorem 2 shows that the resulting convergence rates are jointly determined by the smoothness and intrinsic dimension of the true functions. For example, if the functions h_1 and h_2 have the form in (8) with twice differentiable functions h_{kl}, the convergence rate of the proposed method is of the order $n^{-2/7} \log n$, but the classical nonparametric smoothing methods would yield a slower convergence rate of order $n^{-2/13}$. This demonstrates that deep learning is able to alleviate the curse of dimensionality and identify the low dimensional structure of the data embedded in high dimension space.

4 Experiments

In this section we focus on the numerical performance of DeepEH in comparison with ten competing methods on four real datasets. The Appendix contains more empirical studies and implementation details.

4.1 Evaluation Criteria

Concordance Index. Due to the presence of censoring in survival data, traditional performance measures such as mean squared error cannot be used to evaluate the accuracy of predictions. Instead, the concordance-index (C-index, [22]) is one of the most widely used performance measures for survival models. It assesses how good a model is by measuring the concordance between the rankings of the predicted event times and the true event times. Specifically, if the predicted event time of the i-th individual is $\hat{T}_i$, the C-index is defined by $C = \mathbb{P}(\hat{T}_i < \hat{T}_j | O_i < O_j, \Delta_i = 1)$. However, it is difficult to obtain the predicted event time in most survival models, so the following C-index proposed in [3] is often used in practice:

$$C = \mathbb{P}(\hat{S}(O_i|X_i) < \hat{S}(O_i|X_j) | O_i < O_j, \Delta_i = 1). \tag{11}$$

If $\{X_i, O_i, \Delta_i\}_{i=1}^n$ and $\hat{S}(t|X_i)$ denote observations and predicted conditional probabilities, respectively, the C-index in (11) can be estimated empirically by

$$\hat{C} = \frac{\sum_{i=1}^n \sum_{j=1}^n \Delta_i I(\hat{S}(O_i|X_i) < \hat{S}(O_i|X_j))}{\sum_{i=1}^n \sum_{j=1}^n \Delta_i I(O_i < O_j)}.$$

The range of the C-index is $[0, 1]$, and larger values indicate better performance with a random guess leading to a C-index of 0.5.

Integrated Brier Score. Graf et al. [20] extended the *Brier score* for binary data to right censored data in order to assess the accuracy of an estimated survival function at some time t. Under right censorship, the Brier score at time t is formally given by

$$BS(t) = \frac{1}{n} \sum_{i=1}^n \Big\{ \frac{[\hat{S}(t|X_i)]^2 1(O_i \le t)\Delta_i}{\hat{G}(O_i)} + \frac{[1 - \hat{S}(t|X_i)]^2 1(O_i > t)}{\hat{G}(t)} \Big\},$$

where $\hat{G}$ is the Kaplan-Meier estimate of the survival function. As a global measure, we consider the integrated Brier score (IBS)

$$IBS = \int_0^\tau BS(t)\, dt \tag{12}$$

to evaluate the performance of survival estimates. Here IBS is a proper scoring rule [19], where smaller values of this metric indicate better performance.

4.2 Comparison Methods

We compare our proposed models with three traditional survival methods, random survival forest and six recent deep learning methods.

CoxPH, **AFT**, **EH** represent the standard Cox proportional hazard, accelerated failure time and extended hazard models introduced in Section 2.

RSF leverages the accuracy of random forests for right-censored survival settings [27]. RSF bootstrap samples from the original data and grows a tree for each bootstrapped data set by splitting on nodes that maximize the log-rank difference between their sub-nodes. It then applies the Nelson–Aalen estimator to calculate the cumulative hazard function for each tree and finally averages these estimates.

DeepSurv replaces the linear relationship in the CoxPH model with a neural network [32].

CoxTime extends CoxPH with a time-dependent hazard ratio, learned using a neural network [35].

PCHazard treats the hazard function as piece-wise constant in predefined time-intervals, using a neural network to maximize this log-likelihood [34].

DeepHit directly estimates the joint distribution of events in discrete time, using a multi-task neural network trained with a likelihood loss and a ranking loss [37].

DSM developed a fully parametric method to estimate the competing risks for survival time [41].

DeepAFT is adapted from the proposed DeepEH when enforcing $h_1 = h_2$, and we implement it as a specical case of DeepEH.

Among all the methods considered in this paper, DeepHit is the most general model, but it may only perform well in applications where data are plentiful (which is not the case in existing survival studies) and no lower dimensional structures exist for the survival times.

4.3 Data Sets

Here we evaluate the performance of various survival methods using four real survival datasets: Rotterdam tumor bank and German Breast Cancer Study Group (Rot&GBSG), Molecular Taxonomy of Breast Cancer International Consortium (METABRIC), Worcester Heart Attack Study (WHAS) and Chemotherapy for Stage B/C colon cancer (COLON).

Rot & GBSG is the breast cancer data extracted from the Rotterdam tumor bank [17] and the German Breast Cancer Study Group [47]. The data sets consists of 2,232 patients and 56.7 percent of patients have observed death time. During the studies, researchers recorded the standard features: age, tumor size, number of positive lymph nodes, progesterone and oestrogen receptor status, menopausal status and tumor grade for each patient. In our analysis, tumor size, number of positive lymph nodes and tumor grade are coded to be ordinal features, whereas other features are coded to be continuous.

METABRIC includes paired DNA-RNA profiles and clinical information of 1,980 breast cancer patients, among which 57.7 percent are recorded to die from the cancer [13]. Based on Immunohistochemical 4 score plus Clinical Treatment Score (IHC4+CTS, [36]), DeepSurv [32] considered four genes, four clinical indicators, and each patient's age as features to predict survival. We follow the same choices in our study. The four genes (MK167, EGFG, PGR and ERBB2) and the patient's age are continuous features, while hormone treatment indicator, radiotherapy indicator, chemotherapy indicator, ER-positive indicator are binary features.

WHAS is a data set of 1,638 subjects for studying the factors of acute myocardial infarction [24]. The features contain sex, age, body mass index, left heart failure complications, and order of myocardial infarction, of which age and body mass index are continuous features and others are binary. By the end of the WHAS study, 42.1 percent of the subjects had died of acute myocardial infarction.

COLON was collected from a sample of 929 subjects to evaluate the effects of the drugs levamisole and fluorouracil on resected colon carcinoma [39]. By the end of study, 48.6 percent of patients were dead. Twelve features are included, among which two are continuous features: age, number of lymph nodes with detectable cancer, and ten are categorical features: treatment, sex, obstruction of colon by tumour, perforation of colon, adherence to nearby organs, differentiation of tumour, Extent of local spread, time from surgery to registration, more than 4 positive lymph nodes.

4.4 Results

As in previous work, the evaluation for each dataset is done via 5-fold cross-validation. In each fold, 20% of the training data is reserved as a validation set. The performance is gauged via the C-index and IBS.

Table 1 shows the average C-index and IBS (on test data) and its standard deviation (in parenthesis) over five different train/test splits. In terms of the C-index, DeepEH outperforms the other ten survival methods on three datasets: Rot/GBSG, METABRIC and COLON. For the WHAS dataset, RSF is the best method followed by DeepEH. We also observe that DeepEH improves over its classical predecessor, the EH model. DeepEH improves the performance by more than 6% over EH on Rot/GBSG and METABRIC. In addition, DeepEH outperforms all its submodels, i.e., CoxPH, AFT, DeepSurv and DeepAFT, on all datasets. In terms of IBS, DeepEH performs the best among all eleven survival methods on Rot/GBSG, WHAS and COLON, and it outperformed the other ten methods substantially on Rot/GBSG. Similiar to C-index, DeepEH also has better performance in IBS than its submodels (CoxPH, AFT, EH, DeepSurv and DeepAFT) on all datasets.

Table 1: C-index and IBS achieved on test data by various methods in four datasets. Results listed here are the mean and standard deviation (in parenthesis) of the C-index and IBS over five different train/test splits of each dataset (best performance in bold).

Methods	C-index				IBS			
	Rot/GBSG	METABRIC	WHAS	COLON	Rot/GBSG	METABRIC	WHAS	COLON
CoxPH	0.6638 (0.0111)	0.6332 (0.0131)	0.7598 (0.0120)	0.6489 (0.0411)	0.1853 (0.0046)	0.1879 (0.0126)	0.1492 (0.0051)	0.1989 (0.0115)
AFT	0.7077 (0.0512)	0.6277 (0.0189)	0.7617 (0.0153)	0.6556 (0.0247)	0.1846 (0.0091)	0.1974 (0.0148)	0.2135 (0.0037)	0.1932 (0.0071)
EH	0.6656 (0.0418)	0.6290 (0.0187)	0.8111 (0.0299)	0.6450 (0.0267)	0.1802 (0.0159)	0.1920 (0.0131)	0.1826 (0.0048)	0.1895 (0.0064)
RSF	0.6717 (0.0130)	0.6469 (0.0157)	**0.8694** (0.0179)	0.6558 (0.0210)	0.1766 (0.0037)	**0.1639** (0.0057)	0.1197 (0.0028)	0.1874 (0.0107)
DeepSurv	0.7116 (0.0096)	0.6335 (0.0147)	0.8309 (0.1554)	0.6380 (0.0431)	0.1782 (0.0043)	0.1665 (0.0070)	0.1314 (0.0330)	0.1954 (0.0121)
CoxTime	0.6733 (0.0091)	0.6577 (0.0122)	0.8400 (0.0115)	0.6322 (0.0483)	0.1813 (0.0025)	0.1643 (0.0078)	0.1227 (0.0047)	0.1952 (0.0149)
PCHazard	0.6729 (0.0131)	0.6481 (0.0167)	0.8389 (0.0101)	0.6304 (0.0380)	0.1790 (0.0039)	0.1670 (0.0069)	0.1197 (0.0062)	0.1885 (0.0380)
DeepHit	0.6681 (0.0132)	0.6631 (0.0159)	0.8030 (0.0109)	0.6387 (0.0383)	0.1965 (0.0028)	0.1719 (0.0068)	0.1419 (0.0058)	0.2054 (0.0063)
DSM	0.7359 (0.0630)	0.6173 (0.0167)	0.8330 (0.0347)	0.6494 (0.0436)	0.1840 (0.0020)	0.2187 (0.0047)	0.1306 (0.0117)	0.2009 (0.0143)
DeepAFT	0.7255 (0.0277)	0.6393 (0.0143)	0.8322 (0.0117)	0.6596 (0.0334)	0.1696 (0.0049)	0.1931 (0.0100)	0.1271 (0.0061)	0.1864 (0.0113)
DeepEH	**0.7371** (0.0238)	**0.6690** (0.0129)	0.8416 (0.0108)	**0.6662** (0.0371)	**0.1596** (0.0040)	0.1910 (0.0084)	**0.1173** (0.0032)	**0.1855** (0.0101)

5 Conclusion

Due to its massive potential, deep learning for survival analysis has attracted considerable attention across epidemiology and clinical research. A theoretical understanding, however, is still at its infancy. Mathematically and empirically, this work studied deep extensions of a flexible survival analysis method, the EH model, which can represent a wider family of survival models. We introduced a special but widely used neural network structure to facilitate mathematical study of deep survival analysis. Under the EH model, the neural network estimators of our proposed DeepEH method are asymptotically consistent, as are the corresponding estimators of the survival function. The favorable convergence rates of the new estimators reveal that deep learning is a powerful tool to mitigate the curse of dimensionality with high dimensional covariates. Empirically, the proposed DeepEH model achieves superior C-index and IBS than competing methods. Our methods thus constitute valuable additions to the practical survival modeling toolkit. Their broader impact will likely be to improve estimation accuracy in existing survival analysis applications.

In the future, practical modifications of the neural network architecture may be investigated to adapt the deep survival methods presented here for more complex features such as images [58] or gene expression measurements [8]. While all experiments in this paper were limited to time-invariant features, extensions to time-varying features (longitudinal data) remain of interest. Longitudinal covariates bring rich information to the study but pose challenges to conventional survival models, such as the CoxPH or AFT model, which require continuous measurements of the entire trajectory of the longitudinal covariates that are unavailable in practice [25, 54]. The (bio)statistics community has been actively researching how to best model the survival data and longitudinal covariate processes jointly [14, 46, 57]. In particular, joint modeling for EH models with longitudinal covariates has been explored by Tseng et al. [51, 53]. Thus a natural direction for future work is to extend our DeepEH models to this setting.

References

[1] O. Aalen. Nonparametric inference for a family of counting processes. *The Annals of Statistics*, 6(4):701–726, 1978.

[2] O. Aalen and H. K. Gjessing. Understanding the shape of the hazard rate: A process point of view (with comments and a rejoinder by the authors). *Statistical Science*, 16(1):1–22, 2001.

[3] L. Antolini, P. Boracchi, and E. Biganzoli. A time-dependent discrimination index for survival data. *Statistics in Medicine*, 24(24):3927–3944, 2005.

[4] R. Beran. Nonparametric regression with randomly censored survival data. *Technical report, University of California, Berkeley*, 1981.

[5] N. E. Breslow. Discussion of the paper by DR Cox. *Journal of the Royal Statistical Society, Series B (Statistical Methodology)*, 34(2):216–217, 1972.

[6] P. Chapfuwa, C. Tao, C. Li, C. Page, B. Goldstein, L. Carin, and R. Henao. Adversarial time-to-event modeling. *International Conference on Machine Learning*, 2018.

[7] P. Chapfuwa, C. Tao, L. Carin, and R. Henao. Survival function matching for calibrated time-to-event predictions. *arXiv preprint arXiv:1905.08838*, 2019.

[8] T. Ching, X. Zhu, and L. X. Garmire. Cox-nnet: an artificial neural network method for prognosis prediction of high-throughput omics data. *PLoS Computational Biology*, 14(4): e1006076, 2018.

[9] A. Ciampi and J. Etezadi-Amoli. A general model for testing the proportional hazards and the accelerated failure time hypotheses in the analysis of censored survival data with covariates. *Communications in Statistics-Theory and Methods*, 14(3):651–667, 1985.

[10] A. Cottin, N. Pécuchet, M. Zulian, A. Guilloux, and S. Katsahian. IDNetwork: A deep illness-death network based on multi-states event history process for versatile disease prognostication. In *Survival Prediction-Algorithms, Challenges and Applications*, pages 1–21. Proceedings of Machine Learning Research, 2021.

[11] D. R. Cox. Regression models and life-tables. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 34(2):187–202, 1972.

[12] D. R. Cox. Partial likelihood. *Biometrika*, 62(2):269–276, 1975.

[13] C. Curtis, S. P. Shah, S.-F. Chin, G. Turashvili, O. M. Rueda, M. J. Dunning, D. Speed, A. G. Lynch, S. Samarajiwa, Y. Yuan, et al. The genomic and transcriptomic architecture of 2,000 breast tumours reveals novel subgroups. *Nature*, 486:346–352, 2012.

[14] R. Elashoff, G. Li, and N. Li. *Joint modeling of longitudinal and time-to-event data*. CRC Press, 2016.

[15] J. Etezadi-Amoli and A. Ciampi. Extended hazard regression for censored survival data with covariates: a spline approximation for the baseline hazard function. *Biometrics*, 43(1):181–192, 1987.

[16] D. Faraggi and R. Simon. A neural network model for survival data. *Statistics in Medicine*, 14 (1):73–82, 1995.

[17] J. A. Foekens, H. A. Peters, M. P. Look, H. Portengen, M. Schmitt, M. D. Kramer, N. Brünner, F. Jänicke, M. E. Meijer-van Gelder, and S. C. Henzen-Logmans. The urokinase system of plasminogen activation and prognosis in 2780 breast cancer patients. *Cancer Research*, 60(3): 636–643, 2000.

[18] E. Giunchiglia, A. Nemchenko, and M. van der Schaar. Rnn-surv: A deep recurrent model for survival analysis. In *International Conference on Artificial Neural Networks*, pages 23–32. Springer, 2018.

[19] T. Gneiting and A. E. Raftery. Strictly proper scoring rules, prediction, and estimation. *Journal of the American Statistical Association*, 102(477):359–378, 2007.

[20] E. Graf, C. Schmoor, W. Sauerbrei, and M. Schumacher. Assessment and comparison of prognostic classification schemes for survival data. *Statistics in Medicine*, 18(17-18):2529–2545, 1999.

[21] J. Hao, Y. Kim, T. Mallavarapu, J. H. Oh, and M. Kang. Interpretable deep neural network for cancer survival analysis by integrating genomic and clinical data. *BMC Medical Genomics*, 12 (10):1–13, 2019.

[22] F. E. Harrell, R. M. Califf, D. B. Pryor, K. L. Lee, and R. A. Rosati. Evaluating the yield of medical tests. *JAMA*, 247(18):2543–2546, 1982.

[23] J. Heitz, J. Ficek, M. Faltys, T. M. Merz, G. Rätsch, and M. Hüser. WRSE-a non-parametric weighted-resolution ensemble for predicting individual survival distributions in the ICU. In *Survival Prediction-Algorithms, Challenges and Applications*, pages 54–69. Proceedings of Machine Learning Research, 2021.

[24] D. W. Hosmer Jr, S. Lemeshow, and S. May. *Applied survival analysis: regression modeling of time-to-event data*, volume 618. Wiley, 2008.

[25] F. Hsieh, Y.-K. Tseng, and J.-L. Wang. Joint modeling of survival and longitudinal data: likelihood approach revisited. *Biometrics*, 62(4):1037–1043, 2006.

[26] D. Hubbard, B. Rostykus, Y. Raimond, and T. Jebara. Beta survival models. In *Survival Prediction-Algorithms, Challenges and Applications*, pages 22–39. Proceedings of Machine Learning Research, 2021.

[27] H. Ishwaran, U. B. Kogalur, E. H. Blackstone, M. S. Lauer, et al. Random survival forests. *The Annals of Applied Statistics*, 2(3):841–860, 2008.

[28] A. Kalatian and B. Farooq. Deepsurvival: Pedestrian wait time estimation in mixed traffic conditions using deep survival analysis. *arXiv preprint arXiv:1904.11008*, 2019.

[29] J. D. Kalbfleisch and R. L. Prentice. *The statistical analysis of failure time data*, volume 360. John Wiley & Sons, 2011.

[30] S. Kalmady, W. Sun, J. Ezekowitz, N. Fine, J. Howlett, A. Savu, R. Greiner, and P. Kaul. Improving the calibration of long term predictions of heart failure rehospitalizations using medical concept embedding. In *Survival Prediction-Algorithms, Challenges and Applications*, pages 70–82. Proceedings of Machine Learning Research, 2021.

[31] E. L. Kaplan and P. Meier. Nonparametric estimation from incomplete observations. *Journal of the American Statistical Association*, 53(282):457–481, 1958.

[32] J. L. Katzman, U. Shaham, A. Cloninger, J. Bates, T. Jiang, and Y. Kluger. Deepsurv: personalized treatment recommender system using a Cox proportional hazards deep neural network. *BMC Medical Research Methodology*, 18(1):24, 2018.

[33] N. Keiding, P. K. Andersen, and J. P. Klein. The role of frailty models and accelerated failure time models in describing heterogeneity due to omitted covariates. *Statistics in Medicine*, 16(2): 215–224, 1997.

[34] H. Kvamme and Ø. Borgan. Continuous and discrete-time survival prediction with neural networks. *arXiv preprint arXiv:1910.06724*, 2019.

[35] H. Kvamme, Ø. Borgan, and I. Scheel. Time-to-event prediction with neural networks and Cox regression. *Journal of Machine Learning Research*, 20(129):1–30, 2019.

[36] R. Lakhanpal, I. Sestak, B. Shadbolt, G. M. Bennett, M. Brown, T. Phillips, Y. Zhang, A. Bullman, and A. Rezo. Ihc4 score plus clinical treatment score predicts locoregional recurrence in early breast cancer. *The Breast*, 29:147–152, 2016.

[37] C. Lee, W. R. Zame, J. Yoon, and M. van der Schaar. Deephit: A deep learning approach to survival analysis with competing risks. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[38] E. T. Lee and J. Wang. *Statistical methods for survival data analysis*, volume 476. John Wiley & Sons, 2003.

[39] D. Lin. Cox regression analysis of multivariate failure time data: the marginal approach. *Statistics in Medicine*, 13(21):2233–2247, 1994.

[40] R. G. Miller Jr. *Survival analysis*, volume 66. John Wiley & Sons, 2011.

[41] C. Nagpal, X. R. Li, and A. Dubrawski. Deep survival machines: Fully parametric survival regression and representation learning for censored data with competing risks. *IEEE Journal of Biomedical and Health Informatics*, 2021.

[42] W. Nelson. Theory and applications of hazard plotting for censored failure data. *Technometrics*, 14(4):945–966, 1972.

[43] S. Pölsterl, I. Sarasua, B. Gutiérrez-Becker, and C. Wachinger. A wide and deep neural network for survival analysis from anatomical shape and tabular clinical data. *arXiv preprint arXiv:1909.03890*, 2019.

[44] R. Ranganath, A. Perotte, N. Elhadad, and D. Blei. Deep survival analysis. *arXiv preprint arXiv:1608.02158*, 2016.

[45] K. Ren, J. Qin, L. Zheng, Z. Yang, W. Zhang, L. Qiu, and Y. Yu. Deep recurrent survival analysis. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 4798–4805, 2019.

[46] D. Rizopoulos. *Joint models for longitudinal and time-to-event data: With applications in R*. Chapman and Hall/CRC, 2012.

[47] M. Schumacher, G. Bastert, H. Bojar, K. Huebner, M. Olschewski, W. Sauerbrei, C. Schmoor, C. Beyerle, R. Neumann, and H. Rauschecker. Randomized 2 x 2 trial evaluating hormonal treatment and the duration of chemotherapy in node-positive breast cancer patients. german breast cancer study group. *Journal of Clinical Oncology*, 12(10):2086–2093, 1994.

[48] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15(1):1929–1958, 2014.

[49] T. Sylvain, M. Luck, J. Cohen, H. Cardinal, A. Lodi, and Y. Bengio. Exploring the wasserstein metric for time-to-event analysis. In *Survival Prediction-Algorithms, Challenges and Applications*, pages 194–206. Proceedings of Machine Learning Research, 2021.

[50] A. Tarkhan, N. Simon, T. Bengtsson, K. Nguyen, and J. Dai. Survival prediction using deep learning. In *Survival Prediction-Algorithms, Challenges and Applications*, pages 207–214. Proceedings of Machine Learning Research, 2021.

[51] Y. Tseng, Y. Su, M. Mao, and J. Wang. An extended hazard model with longitudinal covariates. *Biometrika*, 102(1):135–150, 2015.

[52] Y.-K. Tseng and K.-N. Shu. Efficient estimation for a semiparametric extended hazards model. *Communications in Statistics - Simulation and Computation*, 40(2):258–273, 2011.

[53] Y.-K. Tseng, F. Hsieh, and J.-L. Wang. Joint modelling of accelerated failure time and longitudinal data. *Biometrika*, 92(3):587–603, 2005.

[54] A. Tsiatis, V. Degruttola, and M. Wulfsohn. Modeling the relationship of survival to longitudinal data measured with error. applications to survival and CD4 counts in patients with AIDS. *Journal of the American Statistical Association*, 90(429):27–37, 1995.

[55] L.-J. Wei. The accelerated failure time model: a useful alternative to the Cox regression model in survival analysis. *Statistics in Medicine*, 11(14-15):1871–1879, 1992.

[56] A. Xiang, P. Lapuerta, A. Ryutov, J. Buckley, and S. Azen. Comparison of the performance of neural network methods and Cox regression for censored survival data. *Computational Statistics & Data Analysis*, 34(2):243–257, 2000.

[57] C. Xu, P. Z. Hadjipantelis, and J.-L. Wang. Semi-parametric joint modeling of survival and longitudinal data: The r package jsm. *J. Stat. Softw*, 93(2):1–29, 2020.

[58] X. Zhu, J. Yao, and J. Huang. Deep convolutional neural network for survival analysis with pathological images. In *2016 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pages 544–547. IEEE, 2016.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes] In Conclusion section and assumptions section for theory.
   (c) Did you discuss any potential negative societal impacts of your work? [Yes]
   It is possible survival modeling may have some hypothetical negative societal impacts if automation is overly relied upon in critical healthcare decision-making (or some nefarious use by health insurance providers). That said, we do not believe more accurate survival models (of the form introduced here) would have much potential negative impact if simply replacing existing less accurate survival models. We do not believe this work will change somebody's mind about whether or not to use survival modeling for their application in the first place. Thus we have just stated in the Conclusion section that: we believe the broader impact of this work will be to improve estimation accuracy in existing survival analysis applications.
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes]
   (b) Did you include complete proofs of all theoretical results? [Yes]
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes]
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes]
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes] Our results report standard deviations in parenthesis.
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] In Appendix.
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes] We cited them.
   (b) Did you mention the license of the assets? [N/A]
   (c) Did you include any new assets either in the supplemental material or as a URL? [Yes] In the supplemental material.
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]